AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

1. (Currently Amended) A computer system comprising:

at least one processor; and

a flexible operating system executable on the at least one processor to:

determine whether said flexible operating system is being used as a native operating system or as a virtualized operating system on said computer system; and

execute in a first manner as a native operating system on the computer system in response to detecting that said flexible operating system is being used as the native operation system, and execute in a second manner as a virtualized operating system on said computer system in response to detecting that said flexible operating system is being used as the virtualized operating system,

wherein said flexible operating system is configured to operate in a non-virtualized environment when said flexible operating system is being used as the native operating system, and is configured to operate in a virtualized environment when said flexible operating system is being used as the virtualized operating system.

2. (Cancelled)

3. (Currently Amended) The computer system of claim 1 wherein said flexible operating system executing in said second manner comprises said operating system acting as a paravirtualized operating system.

4. (Previously Presented) The computer system of claim 3 wherein said paravirtualized operating system is operable to make a call to a Virtual Machine Monitor (VMM) for performing at least one privileged operation.

5. (Currently Amended) The computer system of claim 1 wherein said flexible operating system determines whether said flexible operating system is being used as the native operating system or the virtualized operating system by:

checking a global variable that indicates whether said flexible operating system is being used as the native operating system or as the virtualized operating system on said computer system.

6. (Currently Amended) The computer system of claim 5, wherein said flexible operating system is executable to further:

execute an instruction which, when the flexible operating system is being used as the virtualized operating system, causes a Virtual Machine Monitor (VMM) to set at least one configuration bit to a first value, and when the flexible operating system is being used as the native operating system, causes the VMM to set said at least one configuration bit to a different value.

7. (Currently Amended) The computer system of claim 6, wherein said flexible operating system is executable to further:

set said global variable based at least in part on the value of said at least one configuration bit after executing said instruction.

8. (Currently Amended) The computer system of claim 1, wherein said flexible operating system is executable to further:

make a call to a Virtual Machine Monitor (VMM) for performing at least one privileged operation.

9. (Previously Presented) The computer system of claim 8 wherein making the call to said VMM uses an Application Program Interface (API) defined for said VMM.

10. (Currently Amended) The computer system of claim 8 wherein making the call to said VMM is used for performing said at least one privileged operation if it is determined that said flexible operating system is being used as virtualized operating system on said computer system.

11. (Currently Amended) A method comprising:

implementing at least one operating system on a computer system;

determining, by said computer system, whether said at least one operating system is a native operating system or a guest operating system on a virtual machine;

said at least one operating system operating in a first manner if determined that it is a native operating system, wherein the native operating system operates in a non-virtualized environment; and

said at least one operating system operating in a second manner if determined that it is a guest operating system on a virtual machine, wherein the guest operating system operates in a virtual environment provided by the virtual machine.

12. (Previously Presented) The method of claim 11 wherein said determining comprises:

said at least one operating system determining whether it is being used as said native operating system or as said guest operating system on the virtual machine.

13. (Currently Amended) The method of claim 12 wherein said at least one operating system determines whether it is being used as said native operating system or as said guest operating system based at least in part on a value of a global variable.

14. (Currently Amended) The method of claim 11 wherein said first manner comprises ~~acting as a~~ said native operating system managing hardware resources of the computer system.

15. (Currently Amended) The method of claim [[11]] 14 wherein said second manner comprises ~~acting as a paravirtualized~~ said guest operating system having access to the computer system hardware resources that are managed by a Virtual Machine Monitor (VMM).

16. (Currently Amended) The method of claim 15 wherein said ~~paravirtualized~~ guest operating system makes, for at least one privileged operation, a call to ~~a Virtual Machine Monitor (VMM)~~ the VMM.

17. (Currently Amended) A computer system comprising:

at least one processor;

a virtual machine monitor (VMM); and

an operating system executable on the at least one processor to:

determine whether said operating system is running as a virtualized operating system or a native operating system; and

adapt operation of said operating system depending on whether it is running as the virtualized operating system or native operating system, wherein the native operating system manages hardware resources in a non-virtualized environment without [[using]] the VMM, and wherein the virtualized operating system manages hardware resources using the VMM.

18. (Previously Presented) The computer system of claim 17 wherein said operating system determines whether said operating system is running as the virtualized operating system or the native operating system by checking the value of a global variable.

19. (Previously Presented) The computer system of claim 18 wherein said operating system checks said value of said global variable before performing certain privileged operations.

20. (Previously Presented) The computer system of claim 17 wherein said operating system performs the determining by determining, before execution of certain privileged instructions, whether said operating system is running as the virtualized operating system or native operating system.

21. (Previously Presented) The computer system of claim 20 wherein said operating systems adapts by if determined that said operating system is running as the virtualized operating system, adapting operation of said operating system in executing said certain privileged instructions.

22. (Previously Presented) The computer system of claim 21 wherein said adapting operation of said operating system in executing said certain privileged instructions comprises:

making at least one call to the VMM.

23. (Previously Presented) The computer system of claim 17 wherein said adapting comprises:

calling the VMM for at least one privileged instruction.

24. (Previously Presented) The computer system of claim 17 wherein said operating system performs the determining by executing an instruction which, when the operating system is being used as the virtualized operating system, causes the VMM to set at least one configuration bit to a first value.

25. (Previously Presented) The computer system of claim 24 wherein said operating system performs the determining by further determining whether said operating system is running as the virtualized operating system or native operating system based at least in part on a determined value of at least one configuration bit after execution of said instruction.

26. (Previously Presented) The computer system of claim 24 wherein said operating system performs the determining by further setting a global variable to a value based at least in part on the value of said at least one configuration bit after execution of said instruction.

27. (Currently Amended) A system comprising:

hardware resources;

a virtual machine monitor (VMM); and

at least one operating system for managing said hardware resources, wherein said at least one operating system is operable to determine whether it is running in a virtualized environment or in a native, non-virtualized environment, wherein said at least one operating system is operable to selectively execute in a first manner if determined that said at least one operating system is running in the native environment and in said second manner if determined that said at least one operating system is running in the virtualized environment, wherein in the first manner said at least one operating system manages said hardware resources without using the VMM, and wherein in the second manner said at least one operating system manages said hardware resources using the VMM.

28. (Cancelled)

29. (Previously Presented) The system of claim 27 wherein said first manner comprises acting as a native operating system.

30. (Previously Presented) The system of claim 27 wherein said second manner comprises acting as a paravirtualized operating system.

31. (Previously Presented) The system of claim 30 wherein said paravirtualized operating system is operable to make a call to the VMM for performing at least one privileged operation.

32. (Cancelled)

33. (Previously Presented) The system of claim 27 wherein said at least one operating system adapts its operation to make a call to said VMM for performance of at least one privileged instruction when said at least one operating system determines that it is running in a virtualized environment.

34. (Cancelled)

35. (Currently Amended) A system comprising:

at least one processor;

a flexible operating system executable on the at least one processor and that is capable of acting as either a native operating system or as a virtualized operating system; and

means for determining whether the flexible operating system is to be used as a native operating system in a non-virtualized environment without a Virtual Machine Monitor (VMM) or as a virtualized operating system in a virtualized environment with the VMM, wherein the determining means stores information that is accessible by the flexible operating system to indicate whether the flexible operating system is being used as a native or as a virtualized operating system.

36. (Original) The system of claim 35 wherein the determining means makes the determination during a boot-up process of the system.

37. (Original) The system of claim 35 further comprising: means for virtualizing resources of said system and multiplexing said resources among one or more virtualized operating systems.

38. (Original) The system of claim 35 wherein said flexible operating system is operable to access the stored information to determine whether said flexible operating system is being used as a native or as a virtualized operating system.

39. (Original) The system of claim 38 wherein if determined that it is being used as a virtualized operating system, said flexible operating system acting as a virtualized operating system.

40. (Original) The system of claim 38 wherein if determined that it is being used as a native operating system, said flexible operating system acting in a first manner, and if determined that it is being used as a virtualized operating system, said flexible operating system acting in a second manner.

41. (Original) The system of claim 40 wherein said first manner comprises acting as a native operating system, and wherein said second manner comprises acting as a paravirtualized operating system.

42. (Currently Amended) The system of claim 35, wherein the virtualized operating system manages hardware resources of the system by using ~~a virtual machine monitor (VMM)~~ the VMM, and wherein the native operating system manages the hardware resources in the non-virtualized environment without using the VMM.

43. (Currently Amended) The computer system of claim 1, wherein the virtualized operating system manages hardware resources of the system by using a virtual machine monitor (VMM), and wherein the native operating system manages the hardware resources in the non-virtualized environment without using the VMM